

О.М. Свінцицька, к.е.н., доц.  
І.В. Пулеко, к.т.н., доц.

Державний університет «Житомирська політехніка»

## Інтеграція Jira, Bitbucket та Sourcetree в системі управління ІТ-проєктами

У статті розглядаються інструменти Atlassian, такі як Jira та Bitbucket в інтеграції з Sourcetree в контексті налаштування автоматизації процесу CI/CD-проєкту вебдодатка. Проведені дослідження показують, що саме така інтеграція відрізняється простотою використання та конфігурацією, що забезпечує інтуїтивний спосіб організації проєктів, управління задачами та налаштування процесу CI/CD в системі управління ІТ-проєктами.

Метою цього дослідження є підвищення ефективності управління, комунікації та співпраці між різними командами розробників, тестувальників та проєктних менеджерів на основі технології автоматизації процесу CI/CD-проєкту вебдодатка на основі інтеграції Jira, Bitbucket та Sourcetree.

Для досягнення поставленої мети в роботі розглянуто практичні кроки по роботі цих інструментів, а також описана технологія налаштування проєкту за допомогою Bitbucket pipelines. Кроки pipelines можуть бути налаштовані на запуск автоматично або вручну, залежно від потреб користувача та конкретного етапу розробки, для різних мов програмування та фреймворків. Щоб налаштувати Pipelines, необхідно створити файл bitbucket-pipelines.yml, де описати кроки процесу CI/CD. Після успішного виконання Bitbucket Pipeline оновлення коду стають доступні в репозиторії, що дає змогу працювати з оновленим кодом. Налаштування автоматизації процесу CI/CD в Jira передбачає автоматичне відстеження пов'язаних процесів збірки та розгортання з проблемами Jira на основі отриманого посилання на ключі проблем у Jira.

**Ключові слова:** управління ІТ-проєктом; Jira; Bitbucket та Sourcetree; CI/CD; Pipelines.

**Постановка проблеми.** В Україні спостерігається швидке зростання інформаційних технологій та ІТ-індустрії. Це призводить до збільшення обсягу ІТ-проєктів, які вимагають ефективного управління, комунікації та співпраці між різними командами розробників, тестувальників та проєктних менеджерів. На ринку представлено багато інструментів для вирішення таких завдань, зокрема найбільш відомими є сервіси від AWS, Atlassian, Azure, кожне з яких має свої переваги і недоліки. Проведені дослідження показують, що саме інтеграція таких інструментів, як Jira, Bitbucket та Sourcetree відрізняється простотою використання та конфігурацією, що надають інтуїтивний спосіб організації проєктів, управління задачами та налаштування процесу CI/CD.

Інструменти Atlassian, такі як Jira, Bitbucket, є добре встановленими і широко використовуваними на ринку. Вони мають велику спільноту користувачів та розробників, що забезпечує доступність підтримки, документації та розширень, системи легко інтегруються з іншими популярними інструментами розробки та управління проєктами. Велика популярність допомагає забезпечити стабільність та надійність системи. Використання цієї системи може бути економічно вигіднішим порівняно з використанням окремих сервісів від AWS або Azure.

Інтеграція Jira, Bitbucket та Sourcetree сприяє впровадженню найкращих практик управління якістю та контролю якості в процес розробки. За допомогою цих інструментів можна встановлювати стандарти кодування, проводити автоматичну перевірку якості, тестування та розгортання програмного забезпечення. Це допомагає знизити кількість помилок, підвищити стабільність продукту та задоволеність клієнтів.

Враховуючи ці особливості та швидкий розвиток ІТ-індустрії в Україні, інтеграція Jira, Bitbucket та Sourcetree стає актуальним, цінним та інноваційним рішенням для управління ІТ-проєктами для компаній будь-якого типу бізнесу. Тому це дослідження спрямоване на розгляд методичних та практичних аспектів налаштування процесу CI/CD вже існуючого проєкту вебдодатка.

Отже, основним завданням цього дослідження є огляд технології налаштування автоматизації процесу CI/CD-проєкту вебдодатка на основі інтеграції Jira, Bitbucket та Sourcetree.

**Аналіз останніх досліджень та публікацій.** Питання управління ІТ-проєктами, зокрема у контексті автоматизації процесу CI/CD-проєкту, розглядається в науковому колі не так широко. Однак питанням управління ІТ-проєктами та інформаційними технологіями приділяється більше уваги. Можна виокремити таких авторів, як А.В. Катренко [5], В.Ф. Гарькава, О.Ю. Єганов, В.М. Бандура, А.М. Арамян [4], Н.І. Шашкова, І.Г. Фадеева, Т.С. Казакова [11], І.О. Башинська [2], Н.В. Морзе [6]. Це свідчить про доцільність поглибленого аналізу вже існуючих технологій автоматизації процесу CI/CD-

проєкту та пошук нових рішень для управління IT-проєктами із застосуванням сучасних інструментів налаштування CI/CD.

**Метою статті** є підвищення ефективності управління, комунікації та співпраці між різними командами розробників, тестувальників та проєктних менеджерів на основі технології автоматизації процесу CI/CD-проєкту вебдодатка на основі інтеграції Jira, Bitbucket та Sourcetree.

**Викладення основного матеріалу.** Jira та Bitbucket – це найбільш відомі продукти, які належать компанії Atlassian Corporation Plc. Це австралійська компанія з розробки програмного забезпечення для IT-індустрії. Більше інформації про продукти Atlassian можна отримати на офіційному сайті компанії [3]. На рисунку 1 відображено основні продукти Atlassian, які застосовуються в процесі розробки програмного забезпечення.



Рис. 1. Atlassian Tools у процесі розробки програмного забезпечення

З наведеного на рисунку 1 процесу бачимо, що основними продуктами, які використовуються в системі управління проєктами розробки програмного забезпечення є: Jira, Confluence та Bitbucket. Jira – інструмент управління проєктами, який дозволяє стежити за завданнями, багами та планувати релізи програмного забезпечення (з 2002 року). Програмне забезпечення Jira може використовуватися як для бізнесу, так і для особистого використання і розраховане переважно на три типи команд: бізнесу, розробки ПЗ та підтримки і обслуговування. Jira також забезпечує зручну інтеграцію з Bitbucket – системою керування версіями (VCS) та репозиторієм Git або Mercurial. Це інструмент для Continuous Integration та спільної роботи розробників над проєктами. Bitbucket дозволяє зберігати та керувати кодовими базами, дозволяє створювати та зливати гілки (branches) коду, робити pull-requests та інші дії, які допомагають команді розробників спільно працювати над проєктом. Крім того, Bitbucket надає можливість здійснювати рецензування коду, відстежування проблем, організацію спільної роботи над проєктами, а також інтеграцію з іншими інструментами [8].

У цьому дослідженні застосовується також графічна програма користувацького інтерфейсу від Sourcetree, яка дозволяє легко працювати з репозиторіями Git. Ця програма, що є клієнтом Git для Windows та Mac, розвиває навички роботи з командним рядком і дає можливість створювати гілки та нові версії, збирати файли, придатні до використання, здійснювати запит для перегляду перед злиттям у головну гілку [10].

Розглянемо практичні кроки по роботі цих інструментів. Передбачається, що вже існує обліковий запис в Jira і відповідний проєкт створено. Поточною задачею є налаштування автоматизації процесу CI/CD в Jira. Для цього треба перейти на головну панель інструментів Jira та в лівій частині зверху обрати продукт Bitbucket. Перейти за рекомендаціями до встановлення і створити робочу область для управління кодом. З Bitbucket можна створювати проблеми для Jira, а також їх редагувати (рис. 2).

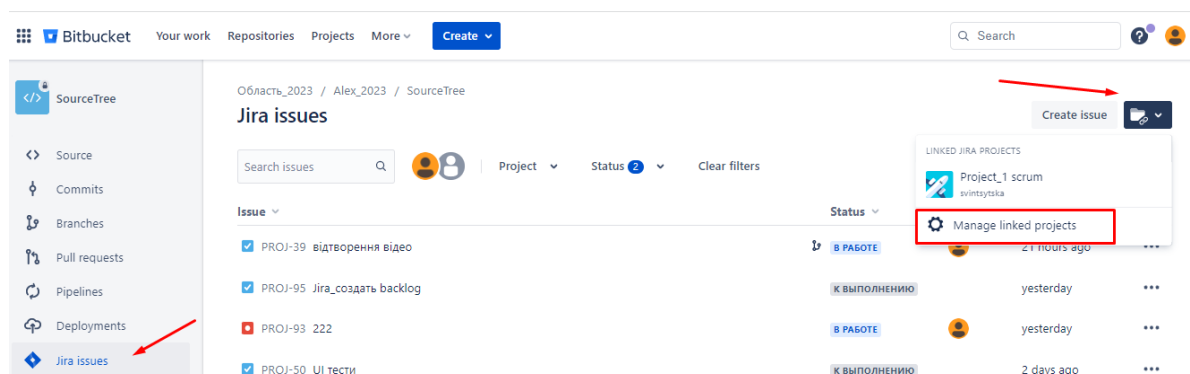


Рис. 2. Відображення проблем для Jira в Bitbucket

У Bitbucket можна створювати commits та pull request, автоматизувати релізи та ін. У свою чергу в Jira можна ці процеси переглядати в самому проекті, створювати гілки та commits і pull request.

Для роботи в Bitbucket створюємо репозиторій під назвою «Веб-додаток». Розширення файлу .git свідчить про відношення до віддаленого (основного) репозиторія в Bitbucket. Цей репозиторій, як правило, є пустим, бо локально з ним не працюють. Отже, наступним кроком у Bitbucket є створення локального репозиторія для роботи та інтеграції його з віддаленим. Для цього спочатку створюємо клон віддаленого репозиторія в графічній програмі Sourcetree. Йдемо за рекомендаціями (рис. 3).

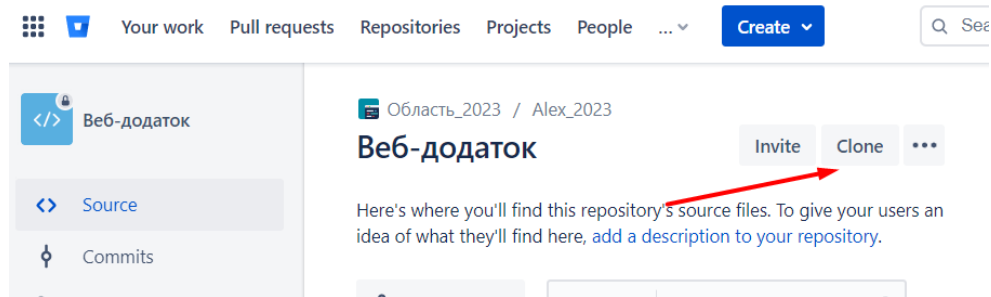


Рис. 3. Створення клону локального репозиторія через Bitbucket для Sourcetree

Також існує інший шлях клонування репозиторія – в Sourcetree. Використовуємо URL віддаленого репозиторія в Bitbucket, який отримуємо від натискання елемента clone. Вказуємо шлях до робочого каталогу і дотримуємося рекомендацій. Sourcetree підтримує безпосередню інтеграцію з багатьма репозиторіями, сервісами хостингу та системами контролю версій. Наприклад, крім Bitbucket, це – GitHub, GitLab та інші. Для синхронізації репозиторіїв використовуємо функцію «Add remote» в Sourcetree. При переході в Bitbucket також відображається синхронізація.

Для налаштувань у Bitbucket використаємо готовий файл проекту вебдодатка з назвою «index» (HTML Document). У робочому просторі Sourcetree в розділі бічної панелі «file status» відображається стан файлів у робочому дереві (файли проекту) та область підготовки, тобто файли, які увійдуть до наступного commit (рис. 4).

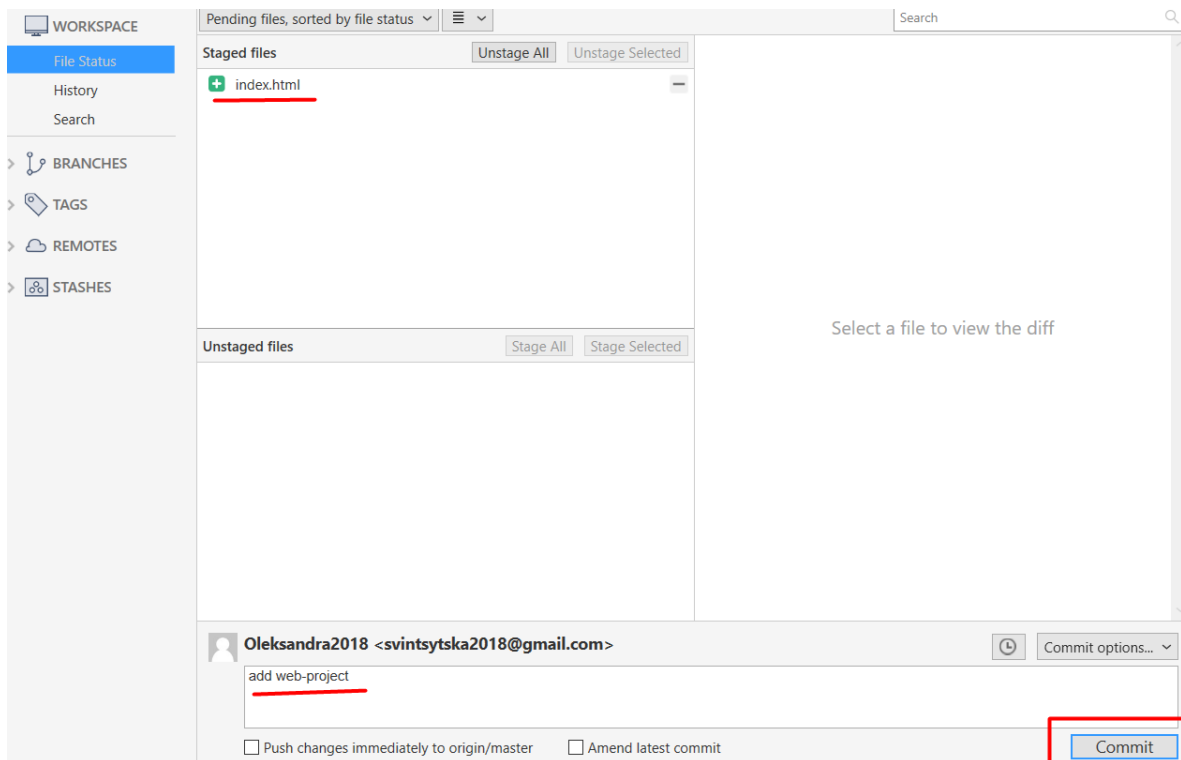


Рис. 4. Робочий простір Sourcetree

У нижньому полі вказується коментар commit (add web-project), який свідчить про зміни, які були внесені в проєкт. За потреби, ставиться галочка навпроти push, що синхронізує цей файл з віддаленим репозиторієм. Це сповіщення стає частиною історії змін. Наступні зміни у робочій папці проєкту будуть відображатися в робочому просторі Sourcetree (рис. 5).

В Jira та Bitbucket можемо створювати та редагувати commits. Якщо гілки не створені, то всі commits автоматично будуть належати основній гілці master. Однак ця гілка призначена, як правило, для тривалого зберігання стабільної версії коду, тому паралельно до основної гілки існують ще робоча та додаткові гілки.

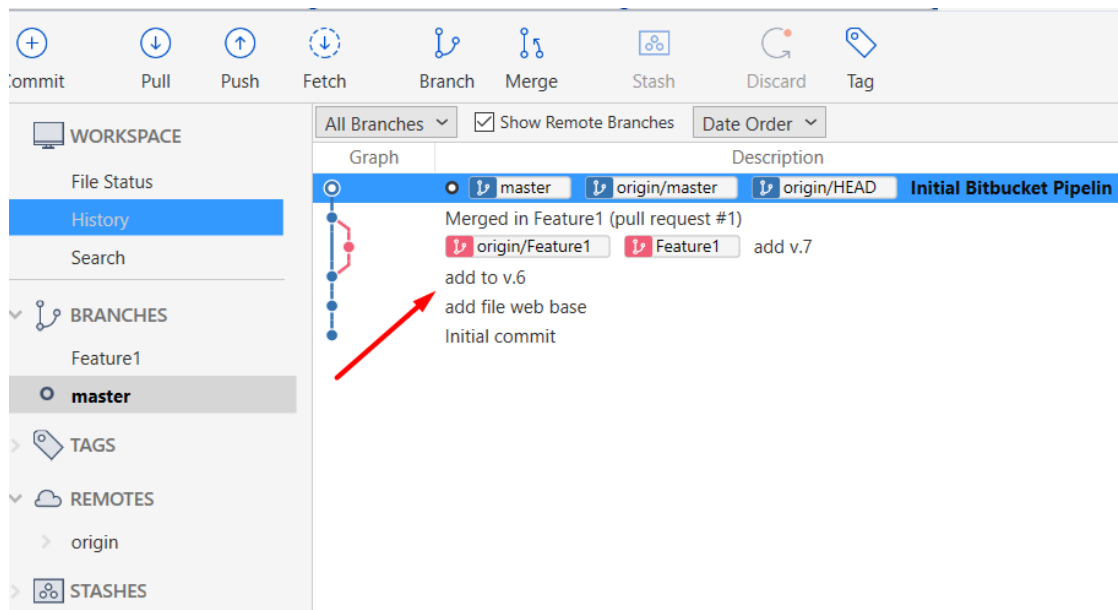


Рис. 5. Огляд робочої області Sourcetree після внесення змін у робочому дереві

У Sourcetree на бічній панелі ми бачимо локальну гілку (master) та віддалену гілку (origin) (рис. 5). Origin – це основна (віддалена) гілка, в якій розміщується вихідний код, від якого починаються інші гілки. Посилання head вказує на останній коміт у поточній гілці. В нашому прикладі створюємо гілку під назвою Feature1. Опція checkout дозволяє переключатися з гілки на гілки, а також синхронізувати дані в робочому дереві. Вона стає незалежною від основної.

Після внесення змін здійснюється відправка даних з локального до віддаленого репозиторія. За допомогою функції push здійснюється копіювання commit з локального репозиторія до віддаленого. При успішному push у них синхронізуються гілки. Всі учасники команди можуть бачити та синхронізувати дані у віддаленому репозиторії.

У Bitbucket в розділі commits також можна бачити синхронізацію. Відповідно до отриманих даних можна здійснити Merge, який виконується в Bitbucket після Pull requests. Pull requests – запит на злиття гілки в проєкт. Pull requests полегшують спільну роботу розробників у Bitbucket. Вони забезпечують зручний вебінтерфейс для обговорення пропонування змін до їх включення до офіційного репозиторія проєкту. Кінцева мета – об'єднати гілки.

У розділі Pull requests є відповідна опція Approve (затвердити), яка передбачає, що інші учасники команди перевіряють код, обговорюють його та вносять зміни. Отримавши запит pull, людина, яка займається підтримкою проєкту, має ухвалити рішення. Якщо гілка готова до використання, можна виконати злиття коду з гілкою main та закрити запит pull. Але якщо у пропонованих змінах є проблеми, можна залишити коментарі у запиті pull. Подальші commits відобразатимуться поряд із відповідними коментарями.

Наступним етапом роботи над проєктом є тестування та налаштування його запуску в Continuous Integration. У Bitbucket є відповідний інструмент – pipelines (конвеєри) [9]. Pipelines – це інтегрована служба CI/CD, вбудована в Bitbucket, що дозволяє автоматично створювати, тестувати та навіть розгортати свій код на основі файлу конфігурації у репозиторії. Для автоматичної збірки треба використовувати ідентифікатори Jira у commits та гілках.

Кроки pipelines можуть бути налаштовані на запуск автоматично або вручну, залежно від потреб користувача та конкретного етапу розробки. Після виконання кожного кроку, pipelines може автоматично переходити до наступного етапу або припинятися у разі помилки чи невдачі.

Pipelines можуть бути налаштовані для різних мов програмування та фреймворків. Вони можуть бути інтегровані з різними інструментами для розробки, такими як GitHub, GitLab, Bitbucket, Jenkins, Travis CI, CircleCI та багато інших. Використання pipelines дозволяє розробникам зосередитися на розробці функціональності програми, не витрачаючи багато часу на ручне тестування та розгортання програмного забезпечення.

Щоб налаштувати Pipelines, необхідно створити файл `bitbucket-pipelines.yml`, де описати кроки процесу CI/CD. Після цього Pipelines автоматично запускається кожен раз, коли ви робите `push` до вашої гілки. Це дозволяє налаштовувати різні етапи в процесі розгортання, враховуючи збірку, тестування та розгортання, використовуючи різні інструменти.

Bitbucket Pipelines містить багато шаблонів на основі різних мов програмування, що дозволяє ручні завдання перетворити на автоматизовані процеси, які можуть надійно та багаторазово виконуватися за допомогою цього сервісу і, зрештою, стають основою для релізів та розгортання всіх робочих середовищ на різних серверах і платформах.

Bitbucket Pipeline дозволяє використовувати будь-які інструменти, які необхідні для запуску проєкту відповідно до потреб. За замовчуванням, конвеєр буде виконувати етапи `build` та `test` на кожен `commit` у вашому репозиторії. Після цього, при кожному `commit` в репозиторії, Bitbucket автоматично запустить конвеєр та виконає всі налаштування [7].

Процес запуску Pipelines (Run) займає декілька хвилин. Після запуску Pipelines відбувається автоматична компіляція, тестування та розгортання програми на вибраному середовищі (за умови налаштування).

Отже, у нашому файлі `bitbucket-pipelines.yml` передбачені такі кроки, як збірка, тестування, літінг та сканування на безпеку, а також розгортання до `staging`- та `production`-середовищ.

Розгортання здійснюється в середовищі `staging`, яке використовується для тестування та перевірки нової версії програмного забезпечення перед тим, як вона буде розгорнута на живому (`production`) сервері. Це іноді називається Pre-Production (передпродакшн) середовищем. Середовище `staging` може бути сконфігуроване як окремий сервер або окремий екземпляр хмарної інфраструктури. Він має бути якомога більш схожим на `production`-середовище з точки зору конфігурації, але зазвичай з меншим обсягом ресурсів та об'ємом даних. Середовище `staging` може бути використане для перевірки функціональності, продуктивності, безпеки та інших аспектів програмного забезпечення перед його випуском у `production`. Якщо тестування проходить успішно, програмне забезпечення може бути розгорнуто на `production`-серверах. Крок «Deployment to Production» передбачає ручний тригер для розгортання коду в `production`. Оскільки досліджуваний проєкт знаходиться на Bitbucket, то можна скористатися будь-якою платформою для розгортання додатка [1].

Основним завданням Deployment є автоматичне розгортання програми на сервері після успішного проходження тестів у Bitbucket Pipelines. За допомогою цієї опції можна проводити відстеження розгортання, щоб переглядати статус розгортань і ключову інформацію про кожен `commit`, яку ви розгортаєте у своїх середовищах. Коли Pipeline завершиться, статус виконання буде відображено на сторінці проєкту в Bitbucket. Успішне виконання Bitbucket Pipeline гарантує, що код буде розгорнуто на обраному середовищі без непередбачених помилок та зменшує ризик виникнення проблем під час розгортання додатка. Приклад успішного запуску наведено на рисунку 6.

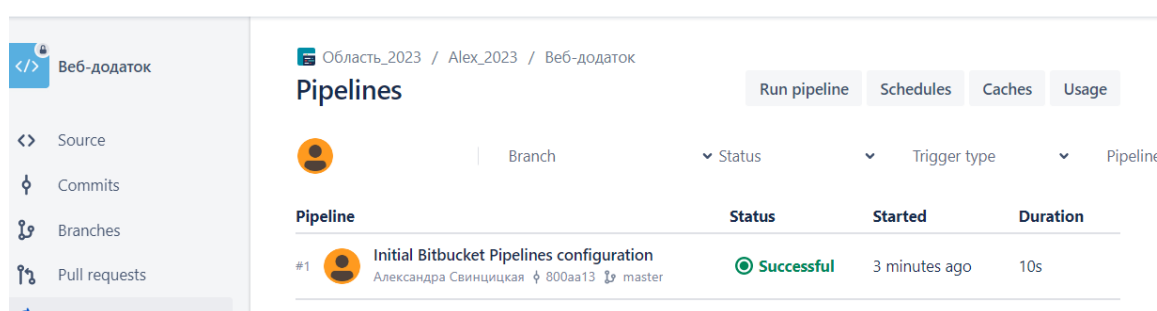


Рис. 6. Приклад успішного запуску Pipeline в Bitbucket

Деталі налаштованого конвеєра по прикладу проєкту «Веб-додаток» виглядають приблизно так, як на рисунку 7. Після успішного виконання Bitbucket Pipeline, оновлення коду будуть доступні в репозиторії, і можна працювати з оновленим кодом.

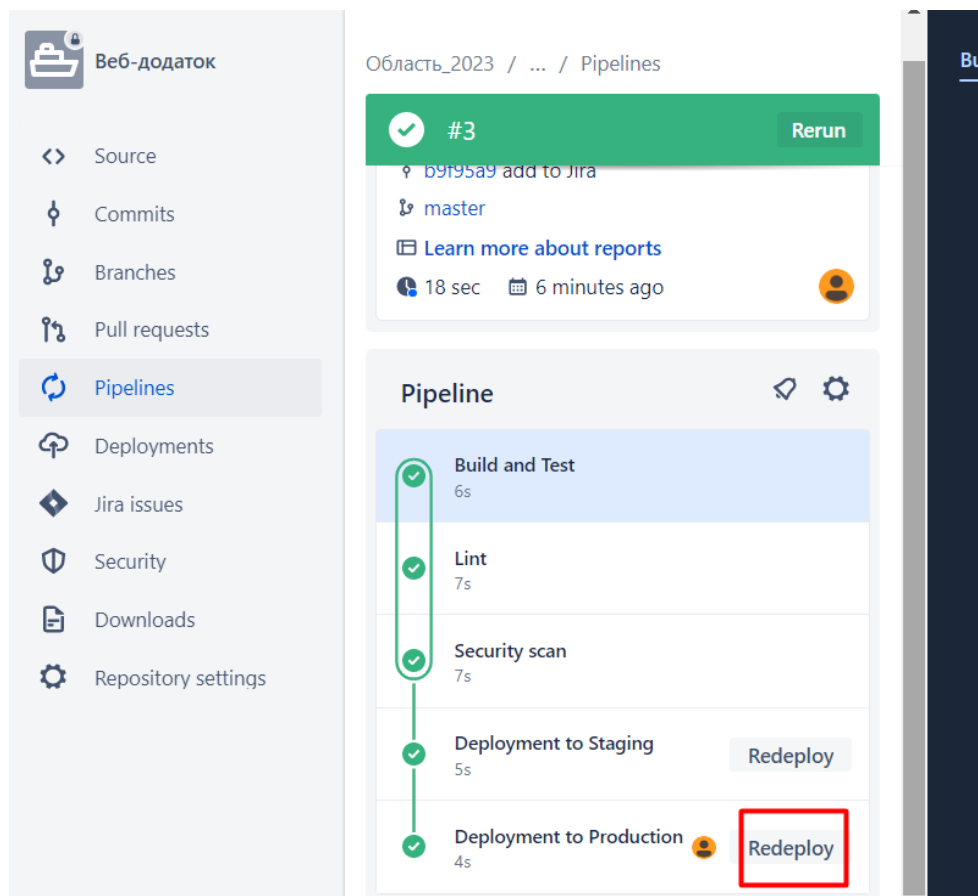


Рис. 7. Етапи успішного налаштування Pipeline в Bitbucket

**Висновки та перспективи подальших досліджень.** Отже, поставлене завдання можна вважати частково виконаним. Для налаштування автоматизації процесу CI/CD в Jira, що передбачає автоматичне відстеження пов'язаних процесів збірки та розгортання з проблемами Jira, залишається отримати посилання на ключі проблем Jira у своїх комітах і гілках. Щоб побачити інформацію про збірку в Jira, треба вставити ключ проблеми в назву гілки. Наприклад, гілка може називатися «feature/ST-1-build-mk2-boosters», і тоді інформація про збірку з'являтиметься у випуску Jira ST-1 кожного разу, коли конвеєр буде працювати з комітом в цій гілці. Щоб переглянути інформацію про розгортання в Jira, треба вставити ключ проблеми в кожне повідомлення commit. Наприклад, ваш commit може називатися «DEVOPS-5 Удосконалення коду селектора», і будь-яке розгортання, яке включає цей коміт, також буде представлено у випуску Jira DEVOPS-5. Таким чином автоматично виявляємо пов'язані проблеми, коли конвеєр створює та розгортає зміни коду. Після налаштування й автоматичної синхронізації інформації про збірку та розгортання з проблемами Jira, буде зручно шукати цю інформацію за допомогою мови запитів Jira (JQL).

#### Список використаної літератури:

1. Deployment (розгортання) за допомогою Bitbucket [Електронний ресурс]. – Режим доступу : <https://support.atlassian.com/bitbucket-cloud/docs/deployments/>.
2. Башинська І.О. Інформаційні технології в управлінні проектами / І.О. Башинська, С.Ю. Каверіна // Економіка і суспільство. – Мукачівський державний університет. – 2017. – № 10. – С. 883–887.
3. Всі продукти Atlassian [Електронний ресурс]. – Режим доступу : <https://www.atlassian.com/ru/software>.
4. Моделювання сучасних комп'ютерних систем / В.Ф. Гарькава, О.Ю. Єганов, В.М. Бандура, А.М. Арамян. – Варшава : RS Global Sp. z O.O., 2020. – 73 с.
5. Катренко А.В. Управління IT-проектами. Книга 1. Стандарти, моделі та методи управління проектами / А.В. Катренко. – Львів : Новий Світ-2000, 2013. – 550 с.
6. Морзе Н.В. Інформаційні системи / Н.В. Морзе, О.З. Піх. – Івано-Франківськ : ЛілеяНВ, 2015. – 383 с.
7. Налаштування Bitbucket Pipelines [Електронний ресурс]. – Режим доступу : <https://support.atlassian.com/bitbucket-cloud/docs/build-test-and-deploy-with-pipelines/>.
8. Ресурси Bitbucket Cloud [Електронний ресурс]. – Режим доступу : <https://support.atlassian.com/bitbucket-cloud/resources/>.

9. Ресурси Bitbucket Pipelines [Електронний ресурс]. – Режим доступу : <https://support.atlassian.com/bitbucket-cloud/docs/get-started-with-bitbucket-pipelines/>.
10. Сайт Sourcetree [Електронний ресурс]. – Режим доступу : <https://www.sourcetreeapp.com/>.
11. Шашкова Н.І. Управління проектами в ІТ сфері: застосування гнучких методологій / Н.І. Шашкова, І.Г. Фадєєва, Т.С. Казакова [Електронний ресурс]. – Режим доступу : [https://zenodo.org/record/5346802#.YkLUredBy\\_I](https://zenodo.org/record/5346802#.YkLUredBy_I).

#### References:

1. *Deployment (rozghortannia) za dopomohoiu Bitbucket*, [Online], available at: <https://support.atlassian.com/bitbucket-cloud/docs/deployments/>
2. BASHYNSKA, I.O. and KAVIERINA, S.Iu. (2017), «Informatsiini tekhnolohii v upravlinni proektamy», *Ekonomika i suspilstvo, Mukachivskiy derzhavnyi universytet*, No. 10, pp. 883–887.
3. *Vsi produkty Atlassian*, [Online], available at: <https://www.atlassian.com/ru/software>
4. HARKAVA, V.F., YECHANOV, O.Yu., BANDURA, V.M. and ARAMIAN, A.M. (2020), *Modeliuvannia suchasnykh kompiuternykh system*, RS Global Sp. z O.O., Varshava, 73 p.
5. KATRENKO, A.B. (2013), *Upravlinnia IT-proektamy. Knyha 1. Standarty, modeli ta metody upravlinnia proektamy*, Novyi Svit-2000, Lviv, 550 p.
6. MORZE, N.V. and PIKH, O.Z. (2015), *Informatsiini systemy*, LileiaNV, Ivano-Frankivsk, 383 p.
7. *Nalashtuvannia Bitbucket Pipelines*, [Online], available at: <https://support.atlassian.com/bitbucket-cloud/docs/build-test-and-deploy-with-pipelines/>
8. *Resursy Bitbucket Cloud*, [Online], available at: <https://support.atlassian.com/bitbucket-cloud/resources/>
9. *Resursy Bitbucket Pipelines*, [Online], available at: <https://support.atlassian.com/bitbucket-cloud/docs/get-started-with-bitbucket-pipelines/>
10. Sait Sourcetree, [Online], available at: <https://www.sourcetreeapp.com/>
11. SHASHKOVA, N.I., FADIEIEVA, I.H. and KAZAKOVA, T.S., *Upravlinnia proektamy v IT sferi: zastosuvannia hnuchkykh metodolohii*, [Online], available at: [https://zenodo.org/record/5346802#.YkLUredBy\\_I](https://zenodo.org/record/5346802#.YkLUredBy_I)

**Свінцицька** Олександра Миколаївна – кандидат економічних наук, доцент, доцент кафедри комп'ютерних наук Державного університету «Житомирська політехніка».

<https://orcid.org/0000-0002-2613-2437>.

Наукові інтереси:

- управління ІТ-проектами;
- інформаційні системи і технології в креативних індустріях.

**Пулєко** Ігор Васильович – кандидат технічних наук, доцент, доцент кафедри комп'ютерної інженерії та кібербезпеки Державного університету «Житомирська політехніка».

<https://orcid.org/0000-0001-8875-017X>.

Наукові інтереси:

- інформаційні системи і технології в управлінні складними технічними системами;
- ІоТ;
- сучасний штучний інтелект.

**Svintsytska O.M., Puleko I.V.**

#### **Integration of Jira, Bitbucket and Sourcetree in the management system of IT projects**

The article examines Atlassian tools such as Jira and Bitbucket in integration with Sourcetree in the context of setting up automation of the CI/CD process of a web application project. Conducted research shows that this kind of integration is characterized by ease of use and configuration, which provides an intuitive way of organizing projects, managing tasks and setting up the CI/CD process in the IT project management system.

The purpose of this study is to improve the efficiency of management, communication and collaboration between different teams of developers, testers and project managers based on technology for automating the CI/CD process of a web application project through the integration of Jira, Bitbucket and Sourcetree.

To achieve the goal, the work considers practical steps in the operation of these tools, as well as describes the technology of setting up the project using Bitbucket pipelines. Pipelines steps can be configured to run automatically or manually, depending on user needs and specific development stage, for different programming languages and frameworks. To set up Pipelines, you need to create a bitbucket-pipelines.yml file that describes the steps of the CI/CD process. After a successful Bitbucket Pipeline execution, code updates are made available in the repository, allowing you to work with the updated code. Setting up CI/CD process automation in Jira involves automatically tracking the associated build and deployment processes with Jira issues based on the received link to issue keys in Jira.

**Keywords:** IT project management; Jira; Bitbucket and Sourcetree; CI/CD; Pipelines.

Стаття надійшла до редакції 07.09.2023.